An Improved Data Aggregation Strategy for Group Recommendations

Toon De Pessemier
iMinds - Ghent University
G. Crommenlaan 8 box 201
B-9050 Ghent, Belgium
Toon.DePessemier@UGent.be

Simon Dooms
iMinds - Ghent University
G. Crommenlaan 8 box 201
B-9050 Ghent, Belgium
Simon.Dooms@UGent.be

Luc Martens
iMinds - Ghent University
G. Crommenlaan 8 box 201
B-9050 Ghent, Belgium
Luc1.Martens@UGent.be

ABSTRACT

Although most recommender systems make suggestions for individual users, in many circumstances the selected items (e.g., movies) are not intended for personal usage but rather for consumption in group. Group recommendations can assist a group of users in finding and selecting interesting items thereby considering the tastes of all group members. Traditionally, group recommendations are generated either by aggregating the group members' recommendations into a list of group recommendations or by aggregating the group members' preferences (as expressed by ratings) into a group model, which is then used to calculate group recommendations. This paper presents a new data aggregation strategy for generating group recommendations by combining the two existing aggregation strategies. The proposed aggregation strategy outperforms each individual strategy for different sizes of the group and in combination with various recommendation algorithms.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Information Filtering; H.5.3 [**Information Interfaces and Presentation**]: Group and Organization Interfaces

General Terms

Algorithms, Experimentation

Keywords

group recommendations, aggregation strategy, combining techniques

1. INTRODUCTION

Although the majority of the currently deployed recommender systems are designed to generate personal suggestions for individual users, in many cases content is selected and consumed by groups of users rather than by individuals. This strengthens the need for group recommendations, providing suggestions thereby considering the tastes of all group members. In the literature, group recommendations have mostly been generated by one of the following two data aggregation strategies [2].

The first aggregation strategy (aggregating recommendations) generates recommendations for each individual user using a general recommendation algorithm. Subsequently, the recommendation lists of all group members are aggregated into a group recommendation list, which (hopefully) satisfies all group members. Different approaches to aggregate the recommendation lists have been proposed during the last decade, such as least misery and plurality voting [7]. Most of them make a decision based on the algorithm's prediction score, i.e., a prediction of the user's rating score for the recommended item. One commonly used way to perform the aggregation is averaging the prediction scores of each member's recommendation list. The higher the average prediction score is, the better the match between the group's preferences and the recommended item.

The second grouping strategy (aggregating preferences) combines the users' preferences into group preferences. This way, the opinions and preferences of individual group members constitute a group preference model reflecting the interests of all members. Again, the members' preferences can be aggregated in different ways, e.g., by calculating the rating of the group as the average of the group members' ratings [7, 1]. After aggregating the members' preferences, the group's preference model is treated as a pseudo user in order to produce recommendations for the group using a traditional recommendation algorithm.

This paper presents a new data aggregation strategy, which combines the two existing strategies and outperforms each of them in terms of accuracy. For both individual data aggregation strategies, we used the average function to combine the individual preferences or recommendations. Although a switching scheme between both aggregation strategies has already been investigated [2], the proposed combined strategy is the first to generate group recommendations by using both aggregation strategies at once, thereby making a more informed decision.

2. EVALUATING GROUP RECOMMENDATIONS

A major issue in the domain of group recommender systems is the evaluation of the accuracy, i.e., comparing the generated recommendations for a group with the true preferences of the group. Performing online evaluations or interviewing groups can be partial solutions but are not feasible

on a large scale or to extensively test alternative configurations. For example, in Section 5, five recommendation algorithms in combination with two data aggregation strategies are evaluated for twelve different group sizes, thereby leading to 120 different setups of the experiment. Therefore, we are forced to perform an offline evaluation, in which synthetic groups are sampled from the users of a traditional single-user data set. Since movies are often watched in group, we used the MovieLens (100K) data set for this evaluation.

In the literature, group recommendations have been evaluated several times by using a data set with simulated groups of users. Baltrunas et al. [1] used the MovieLens data set to simulate groups of different sizes (2, 3, 4, 8) and different degrees of similarity (high, random) with the aim of evaluating the effectiveness of group recommendations. Chen et al. [4] also used the MovieLens data set and simulated groups by randomly selecting the members of the group to evaluate their proposed group recommendation algorithm. They simulated group ratings by calculating a weighted average of the group members' ratings based on the users' opinion importance parameter. Quijano-Sánchez et al. [8] used synthetically generated data to simulate groups of people in order to test the accuracy of group recommendations for movies. In addition to this offline evaluation, they conducted an experiment with real users to validate the results obtained with the synthetic groups. One of the main conclusions of their study was that it is possible to realize trustworthy experiments with synthetic data, as the online user test confirmed the results of the experiment with synthetic data. This conclusion justifies the use of an offline evaluation with synthetic groups to evaluate the group recommendations in our experiment.

This offline evaluation is based on the traditional procedure of dividing the data set in two parts: the training set, which is used as input for the algorithm to generate the recommendations, and the test set, which is used to evaluate the recommendations. In this experiment, we ordered the ratings chronologically and assigned the oldest 60% to the training set and the most recent 40% to the test set, as this reflects a realistic scenario the best.

The used evaluation procedure was adopted from Baltrunas et al. [1] and is performed as follows. Firstly, synthetic groups are composed by selecting random users from the data set. All users are assigned to one group of a predefined size. Secondly, group recommendations are generated for each of these groups based on the ratings of the members in the training set. Since group recommendations are intended to be consumed in group and to suit simultaneously the preferences of all members of the group, all members receive the same recommendation list. Thirdly, since no group ratings are available, the recommendations are evaluated individually as in the classical single-user case, by comparing (the rankings of) the recommendations with (the rankings of) the items in the test set of the user using the Normalized Discounted Cumulative Gain (nDCG) at rank 5. The nDCG is a standard information retrieval measure, used to evaluate the recommendation lists [1].

3. RECOMMENDATION ALGORITHMS

The effectiveness of the different aggregation strategies is measured for different sizes of the group and in combination with various state-of-the art recommendation algorithms. The used implementation of **Collaborative Filtering** (CF) is based on the work of Breese et al [3]. This nearest neighbor CF uses the Pearson correlation metric for discovering similar users in the user-based approach (UBCF) or similar items in the item-based approach (IBCF) based on the rating behavior of the users. As **Content-Based recommender** (CB) the InterestLMS predictor of the open source implementation of the Duine framework [9] is adopted (and extended to consider extra metadata attributes). Based on the actors, directors, and genres of the content items and the user's ratings for these items, the recommender builds a profile model for every user. This profile contains an estimation of the user's preference for each genre, actor, and director that is assigned to a rated item, and is used to predict the user's preference for unseen media items by matching the metadata of the items with the user's profile. The used **hybrid recommender** (Hybrid) combines the recommendations with the highest prediction score of the IBCF and the CB recommender into a new recommendation list. The result is an alternating list of the best recommendations originating from these two algorithms. A user-centric evaluation comparing different algorithms based on various characteristics showed that this straightforward combination of CF and CB recommendations outperforms both individual algorithms on almost every qualitative metric [6]. As recommender based on matrix factorization, we opted for the open source implementation of the **SVD recommender** (SVD) of the Apache Mahout project [10]. This recommender is configured to use 19 features, which equals the number of genres in the MovieLens data set, and the number of iterations is set at 50. To compare the results of the various recommenders, the **popular recommender** was introduced as a baseline. This recommender generates for every user always the same list of most-popular items, which is based on the number of received ratings and the mean rating of each item.

4. COMBINING STRATEGIES

Previous research [5] has shown that the used aggregation strategy in combination with the recommendation algorithm has a major influence on the accuracy of the group recommendations. Certain algorithms (such as CB and UBCF) produce more accurate group recommendations when the aggregating preferences strategy is used, whereas other algorithms (such as IBCF and SVD) obtain a higher accuracy in combination with the aggregating recommendations strategy. So, the choice of the aggregation strategy is crucial for each algorithm in order to obtain the best group recommendations. Instead of selecting one individual aggregation strategy, traditional aggregation strategies can be combined with the aim of obtaining group recommendations which outperform the group recommendations of each individual aggregation strategy. In this context, Berkovsky and Freyne [2] witnessed that the aggregating recommendations strategy yields a lower MAE (Mean Absolute Error) than the aggregating preferences strategy if the user profiles have a low density (i.e., containing a low number of consumptions). In contrast for high-density profiles, the aggregating preferences strategy resulted in the lowest MAE, thereby outperforming the aggregating recommendations strategy in terms of accuracy. Therefore, Berkovsky and Freyne proposed a switching scheme based on the profile density, which yielded a small accuracy improvement compared to the individual strategies. However, their results were obtained in

a very specific setting. They only considered the accuracy of recommendations generated by a CF algorithm, the MAE metric was used to estimate the accuracy, and they focused on the specific use case of recipe recommendations using a rather small data set (approximately 3300 ratings). Because of these specific settings, we were not able to obtain an accuracy improvement by using such a switching scheme on the MovieLens data set.

Therefore, we propose an advanced data aggregation strategy which combines both individual aggregation strategies thereby yielding an accuracy gain compared to each individual aggregation strategy for different recommendation algorithms. This combination of strategies aggregates the preferences of the users as well as their recommendations with the aim of merging the knowledge of the two aggregation strategies into a final group recommendation list. The idea is that if one of the aggregation strategies comes up with a less suitable or undesirable group recommendation, the other aggregation strategy can correct this mistake. This makes the group recommendations resulting from the combination of strategies more robust than the group recommendations based on a single aggregation strategy.

The two aggregation strategies are combined as follows. First, group recommendations are calculated by using the selected recommendation algorithm and the aggregating preferences strategy. The result is a list of all items, ordered according to their prediction score. In case of an individual aggregation strategy, the top-N items on that list are selected as suggestions for the group. After calculating the group recommendations using the aggregating preferences strategy, or in parallel with it, group recommendations are generated using the chosen algorithm and the aggregating recommendations strategy. Again, the result is an ordered list of items with their corresponding prediction score.

Both of these lists with group recommendation can still contain items that are less suitable for the group, even at the top of the list. The next phase will try to eliminate these items by comparing the two resulting recommendation lists. Items that are at the top of both lists are probably interesting recommendations, whereas items at the bottom of both lists are usually less suitable for the group. Less certainty exists about the items that are at the top of the recommendation list that is generated by one of the aggregation strategies but that are in the middle or even at the bottom of the recommendation list produced by using the other aggregation strategy. Therefore, both recommendation lists are adapted by eliminating these uncertain items in order to contain only items that appear at the top of both recommendation lists, thereby reducing the risk of recommending undesirable or less suitable items to the group. So, items that are ranked below a certain threshold position in (at least) one of the recommendation lists generated by the two aggregation strategies, are removed from both lists. If only one aggregation strategy is used, identifying uncertain items based on the results of a complementary recommendation list is not possible. In this experiment, we opted to exclude these items from the recommendation lists, that are not in the top-5% of both recommendation lists (i.e., the top-84 of recommended items for the MovieLens data set). As a result, the recommendation lists contains only items that are identified as 'the most suitable' by both aggregation strategies, ordered according to the prediction scores calculated using either the aggregating preferences strategy or the aggregating recommendations strategy.

Subsequently, the two recommendation lists are combined into one recommendation list by combining the prediction scores of each aggregation strategy per item. In this experiment, we opted for the average as method to combine the prediction scores. So in the resulting recommendation list, each item's prediction score is the average of the item's prediction score generated by the aggregating preferences strategy and the item's prediction score produced by the aggregating recommendations strategy. Alternative combining methods are also possible, e.g., a weighted average of the prediction scores with weights depending on the performance of each individual aggregation strategy. Then, the items are ordered by their new prediction score in order to obtain the final list of group recommendations.

5. RESULTS

Our combined aggregation strategy is compared to the individual aggregation strategies in Figure 1. Since users are randomly combined into groups and the accuracy of group recommendations is depending on the composition of the groups, the accuracy slightly varies for each partitioning of the users into groups. (Except for the partitioning of the users into groups of 1 member, which is only possible in 1 way.) Therefore, the process of composing groups by taking a random selection of users is repeated 30 times and just as much measurements of the accuracy are performed. So, the graph shows the mean accuracy of these measurements as an estimation of the quality of the group recommendations (on the vertical axis), as well as the 95% confidence interval of the mean value, in relation to the recommendation algorithm, aggregation strategy, and the group size. The group size is indicated on the horizonal axis. The vertical axis crosses the horizontal axis at the quality level of the most-popular recommender. The prefix "Combined" of the bar series stands for the proposed aggregation strategy which combines the aggregating preferences and aggregating recommendations strategy. The prefix "Pref" and "Rec" indicate the accuracy of the two individual strategies, respectively the aggregating preferences and aggregating recommendations strategy. For each algorithm, only the most accurate individual strategy is shown: aggregating preferences for UBCF and CB, aggregating recommendations for SVD, IBCF, and Hybrid [5].

The non-overlapping confidence intervals indicate a significant improvement of the combined aggregation strategy compared to the best individual aggregation strategy. Table 1 shows the results of the statistical T-tests comparing the mean accuracy of the recommendations generated by the best individual aggregation strategy and by the combined aggregation strategy for groups with size = 5. (Similar results are obtained for other group sizes.) The null hypothesis, H_0 = the mean accuracy of the recommendations generated by using the best individual aggregation strategy is equal to the mean accuracy of the recommendations generated by using the combined aggregation strategy. The small p-values (all smaller than 0.05) prove the significant accuracy improvement of our proposed aggregation strategy.

6. CONCLUSIONS

This paper presents a new strategy to aggregate the tastes of multiple users in order to generate group recommenda-

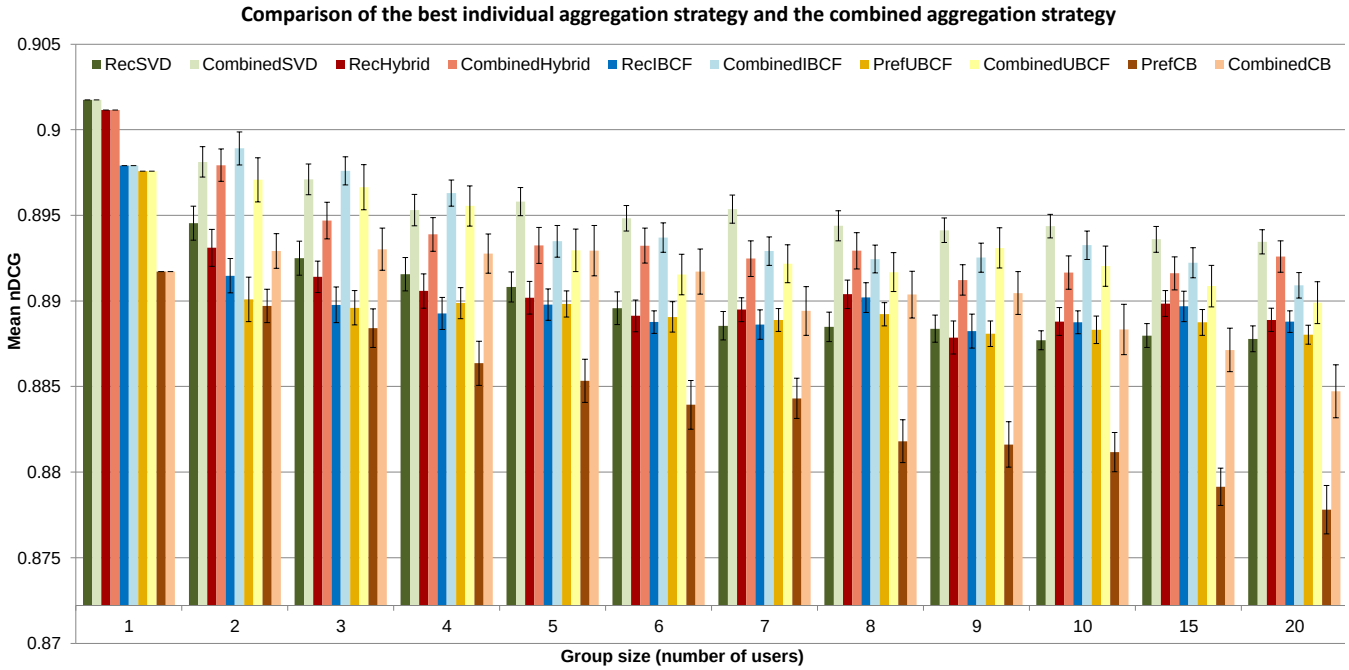

Figure 1: The accuracy of the group recommendations calculated using the best individual aggregation strategy and the combined aggregation strategy

Table 1: Statistical T-test comparing the best individual aggregation strategy and the combined aggregation strategy for groups with size=5

Algorithm	t(58)	p-value
SVD	-4.39	**0.00**
Hybrid	-2.53	**0.01**
IBCF	-2.33	**0.02**
UBCF	-2.66	**0.01**
CB	-3.55	**0.00**

tions. Both existing data aggregation strategies are combined to make a more informed decision hereby reducing the risk of recommending undesirable or less suitable items to the group. The results show that the combination of aggregation strategies outperforms the individual aggregation strategies for various sizes of the group and in combination with various recommendation algorithms. The proposed aggregation strategy can be used to increase the accuracy of (commercial) group recommender systems.

7. REFERENCES

[1] L. Baltrunas, T. Makcinskas, and F. Ricci. Group recommendations with rank aggregation and collaborative filtering. In *Proceedings of the fourth ACM conference on Recommender systems*, RecSys '10, pages 119–126, New York, NY, USA, 2010. ACM.

[2] S. Berkovsky and J. Freyne. Group-based recipe recommendations: analysis of data aggregation strategies. In *Proceedings of the fourth ACM conference on Recommender systems*, RecSys '10, pages 111–118, New York, NY, USA, 2010. ACM.

[3] J. S. Breese, D. Heckerman, and C. Kadie. Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence*, UAI'98, pages 43–52, San Francisco, CA, USA, 1998.

[4] Y.-L. Chen, L.-C. Cheng, and C.-N. Chuang. A group recommendation system with consideration of interactions among group members. *Expert Systems with Applications*, 34(3):2082 – 2090, 2008.

[5] T. De Pessemier, S. Dooms, and L. Martens. Design and evaluation of a group recommender system. In *Proceedings of the sixth ACM conference on Recommender systems*, RecSys '12, pages 225–228, New York, NY, USA, 2012. ACM.

[6] S. Dooms, T. De Pessemier, and L. Martens. A user-centric evaluation of recommender algorithms for an event recommendation system. In *Proceedings of the workshop on User-Centric Evaluation of Recommender Systems and Their Interfaces at ACM Conference on Recommender Systems (RECSYS)*, pages 67–73, 2011.

[7] J. Masthoff. Group modeling: Selecting a sequence of television items to suit a group of viewers. *User Modeling and User-Adapted Interaction*, 14:37–85, 2004.

[8] L. Quijano-Sanchez, J. A. Recio-Garcia, and B. Diaz-Agudo. Personality and social trust in group recommendations. In *Proceedings of the 2010 22nd IEEE International Conference on Tools with Artificial Intelligence - Volume 02*, ICTAI '10, pages 121–126, Washington, DC, USA, 2010. IEEE Computer Society.

[9] Telematica Instituut / Novay. Duine Framework, 2009. Available at `http://duineframework.org/`.

[10] The Apache Software Foundation. Apache Mahout, 2012. Available at `http://mahout.apache.org/`.